In The Claims

1-99. (Cancelled)

100. (Currently Amended) A computer system, comprising:

a bus;

a central processing unit coupled to said bus; and

a graphics accelerator coupled to said bus, said graphics accelerator including a texture cache system, said texture cache system including a texture cache memory that stores texels to be used by a texel value generating circuit, a cache controller that performs a replacement policy determination for texel data to be stored in said texture cache memory, and a direct memory access engine that retrieves texel data from memory, [wherein said replacement policy determination operates such that a common priority scheme is applied to a plurality of cache lines containing texels] wherein said cache controller includes a first set of flags and a second set of flags, each of the first set of flags and the second set of flags having a bit position for each of a plurality of cache lines in the cache memory, wherein the cache controller is configured to set a bit position in the first set of flags for each cache line that will be used to render a first triangle, wherein the cache controller is further configured to set a bit position in the second set of flags for each cache line that will be used to render a second triangle, and wherein the cache controller is configured to only replace texels in cache lines for which neither the bit position associated with the first set of flags nor the bit position associated with the second set of flags has been set.

101-103. (Cancelled)

2

104. (Original) A computer system, comprising:

a bus;

a central processing unit coupled to said bus; and

a graphics accelerator coupled to said bus, said graphics accelerator including a texture cache system, said texture cache system including a texture cache memory that stores texels to be used by a texel value generating circuit, wherein said texture cache system is capable of operating in a prefetch mode such that during the rendering of a first polygon, a set of texels including at least those texels needed for completely rendering a second polygon are prefetched and stored in said texture cache memory, and wherein said set of texels are prefetched if it is determined that said set of texels can fit into one half of said texture cache memory.

3

105. (Currently Amended) A method for processing texels, comprising:

caching a first plurality of texel values in a cache memory;

generating a first plurality of pixel texture values using the first plurality of cached texel values, said generating comprising reusing at least one of the first plurality of cached texel values;

defining a texture over a first triangle using the first plurality of pixel texture values;

determining whether a second plurality of texel values can fit into one half of the cache memory;

caching [a] the second plurality of texel values in the cache memory if it is determined that the second plurality of texel values can fit into one half of the cache memory[, wherein the second plurality of texel values is less in number than the first plurality of texel values]; and

generating a second plurality of pixel texture values using at least one of the first plurality of cached texel values and at least one of the second plurality of cached texel values.

27

6

4 3
106. (Original) The method of claim 105, further comprising defining a texture over a second triangle using the second plurality of pixel texture values.

5 3
107. (Original) The method of claim 105, wherein caching the first plurality of texel values includes prefetching only the texel values necessary for defining the texture over the first triangle.

6 3
108. (Original) The method of claim 105, wherein generating each of the first plurality of pixel texture values includes reading four texel values of the first plurality of cached texel values.

61 109-110. (Cancelled)

7
111. (Currently Amended) A method for processing textures, comprising:

storing a plurality of texture maps in a main memory;

transferring a subset of said plurality of texture maps from the main memory to a local memory of a graphics accelerator;

accessing texels from the subset of said plurality of texture maps in the local memory; and

caching the texels in the graphics accelerator, wherein accessing texels includes reading page table entries in the local memory, the page table entries providing physical addresses related to said plurality of texture maps.

112. (Cancelled)

28

G